National University of Sciences and Technology (NUST)

SEECS

Digital Image Processing

Bit Plane Slicing

Steganography

Bit Plane Slicing

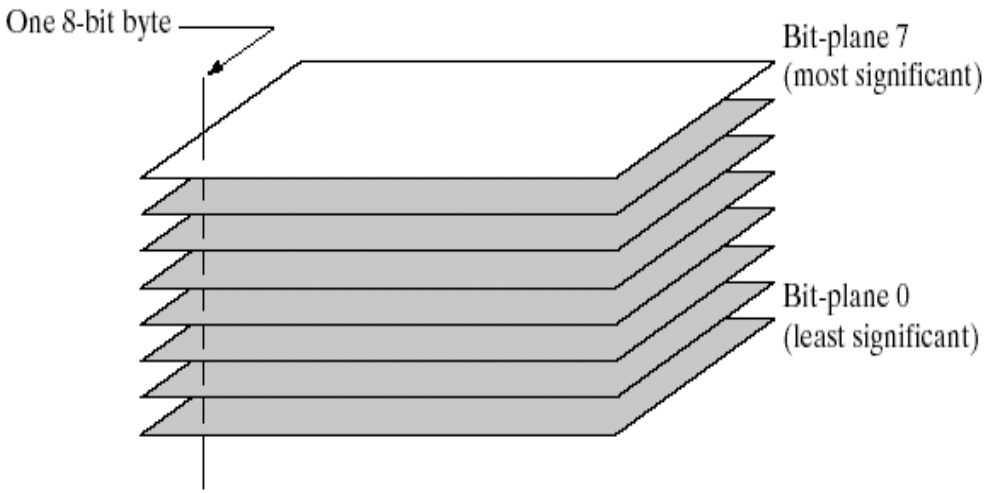
Objective:

Highlights the contribution made to the image appearance by specific bits.

Suppose an image is of 8 bits i.e. each pixel is represented by 8 bits.

Higher-order bits contain the majority of the visually significant data.

Useful for analyzing the relative importance played by each bit of the image. It is useful in compression.



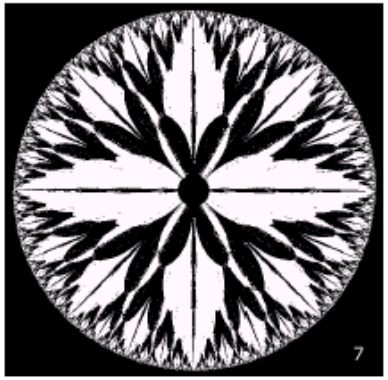
Bit Plane Slicing

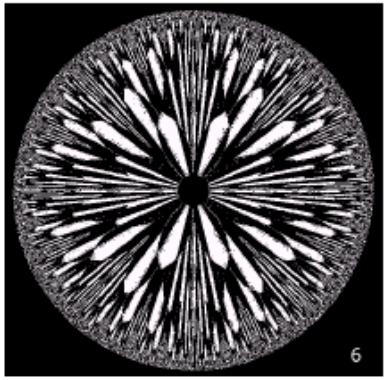
(MSB) 11111010 (LSB)

250 (11111010)	126(01111110)	26 (00011010)	255 (11111111)
0 (00000000)	42 (00101010)	32 (00100000)	21 (00010101)
1 (00000001)	2 (00000010)	16 (00010000)	22 (00010110)
99 (01100011)	198 (11000110)	8 (00001000)	96 (01100000)

Bit Plane Slicing

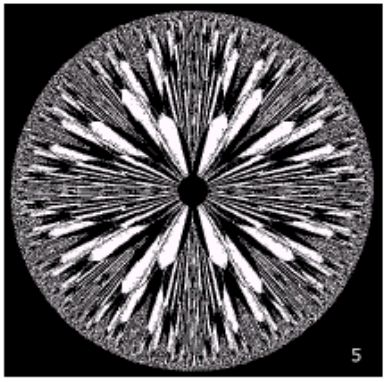
[10000000]

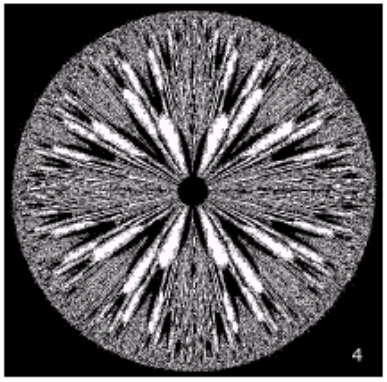


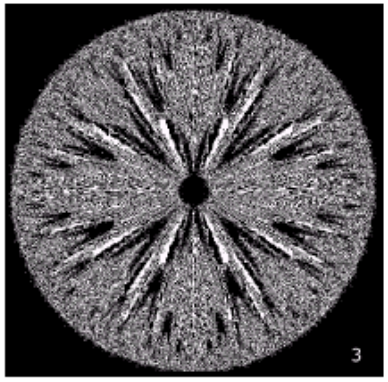


[01000000]

[00100000]

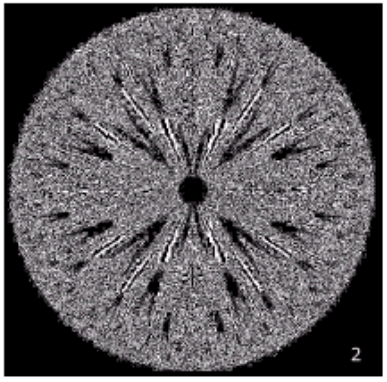


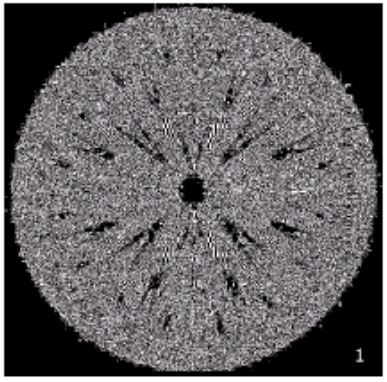


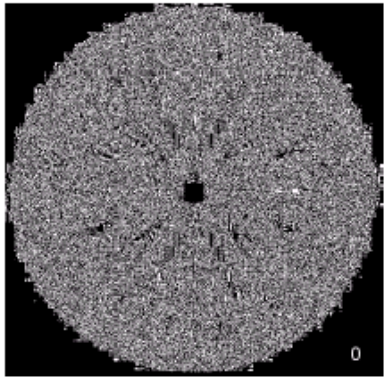


[00001000]

[00000100]







[00000001]

Bit Plane Slicing

NUST

Bit planes 1 through 8

Bit Plane Slicing

Reconstructed image using only bit planes 8 and 7

Reconstructed image using only bit planes 8, 7 and 6

Reconstructed image using only bit planes 7, 6 and 5

Steganography

The science of writing hidden messages in such a way that no one, apart from the sender and intended recipient, suspects the existence of the message.

Steganography

The main goal of steganography is to hide a message m in some audio or video (cover) data d, to obtain new data d', practically indistinguishable from d, by people, in such a way that an eavesdropper cannot detect the presence of m in d'.

The main goal of watermarking is to hide a message m in some audio or video (cover) data d, to obtain new data d', practically indistinguishable from d, by people, in such a way that an eavesdropper cannot remove or replace m in d'.

www.cs.bham.ac.uk/~mdr/teaching/modules03/.../Steganography.pdf

Steganography

Differences between steganography and watermarking are both subtle and essential.

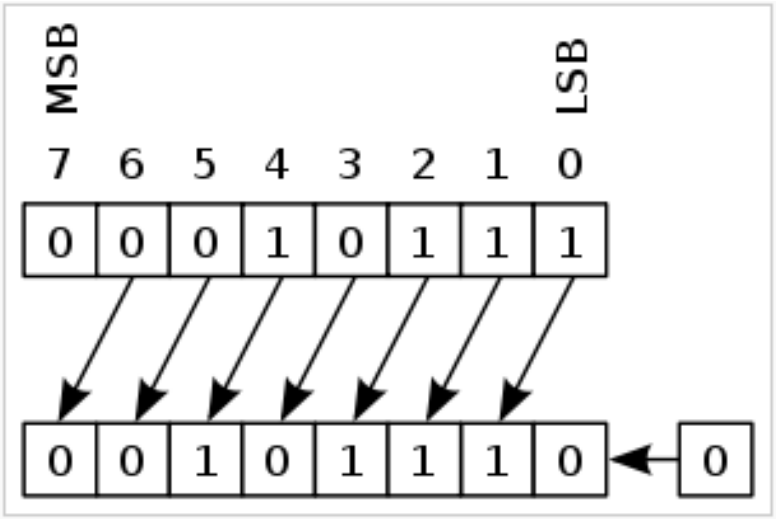
Shortly, one can say that watermarking is about protecting the content of messages, steganography is about concealing its very existence.

Steganography methods usually do not need to provide strong security against removing or modification of the hidden message. Watermarking methods need to be very robust to attempts to remove or modify a hidden message.

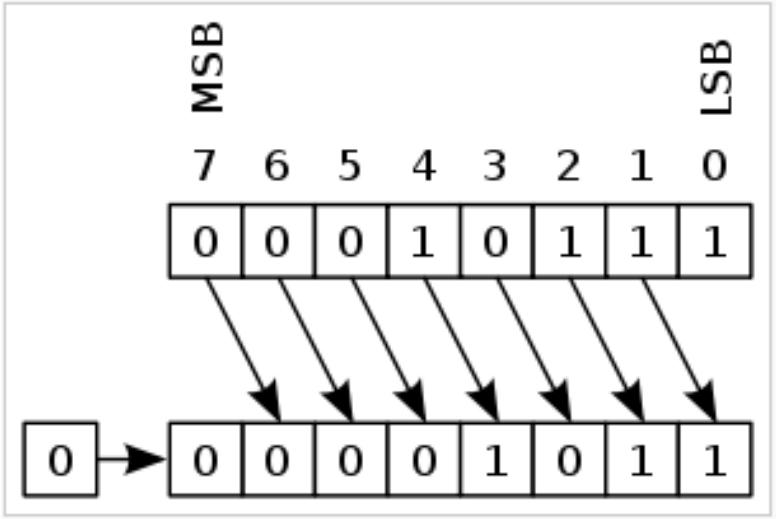
www.cs.bham.ac.uk/~mdr/teaching/modules03/.../Steganography.pdf

Steganography

- Before Moving On ….
- Recall – Logical Shift Operators



Logical left shift one bit



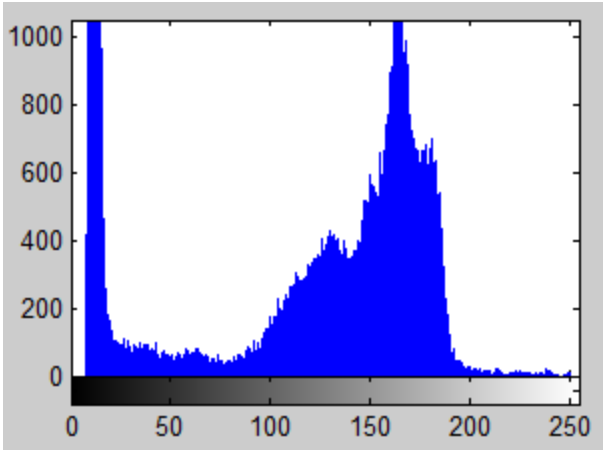
Logical right shift one bit

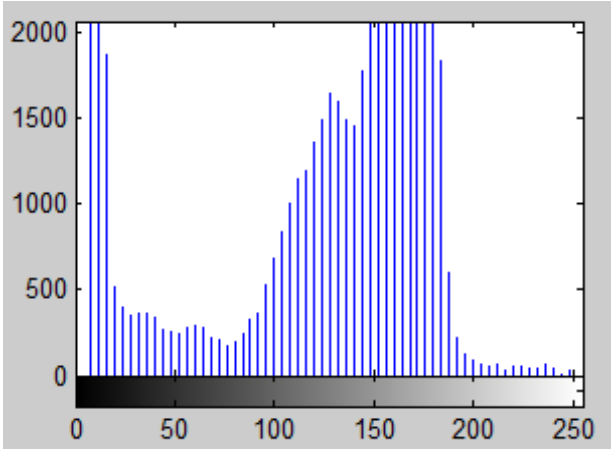
Steganography

8-bit Image

6-bit Image

Two least significant bits are 0

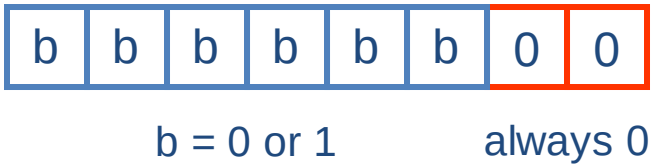




If an image is quantized, say from 8 bits to 6 bits and redisplayed it can be difficult to tell the difference between the two images.

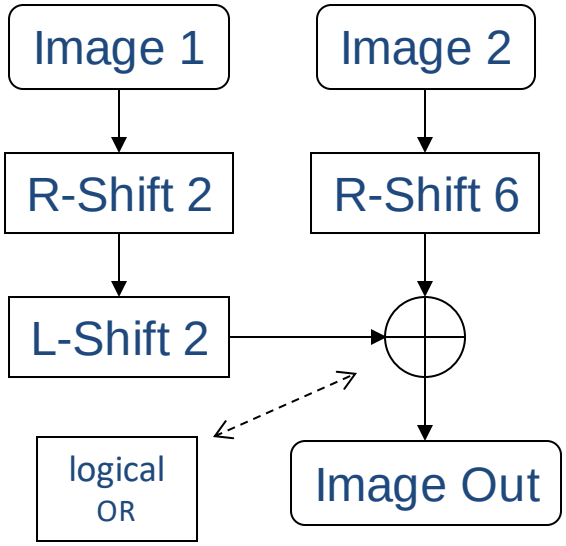
Steganography

If the 6-bit version is displayed as an 8-bit image then the 8-bit pixels all have zeros in the lower 2 bits:



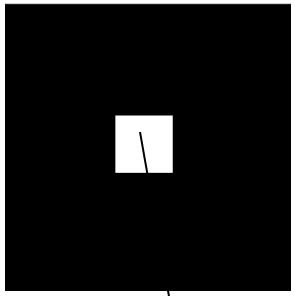
This introduces the possibility of encoding other information in the low-order bits.

That other information could be a message, perhaps encrypted, or even another image.



X-Shift *n* = logical left or right shift by *n* bits.

Steganography



182

1	0	1	1	0	1	1	0

R-Shift 2

0	0	1	0	1	1	0	1

L-Shift 2

180

1	0	1	1	0	1	0	0

220

1	1	0	1	1	1	0	0

R-Shift 6

03

0	0	0	0	0	0	1	1

⊕

183

1	0	1	1	0	1	1	1

Steganography

How to get the second image

183

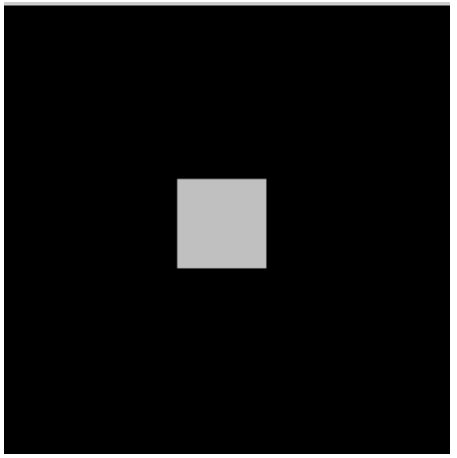
1	0	1	1	0	1	1	1

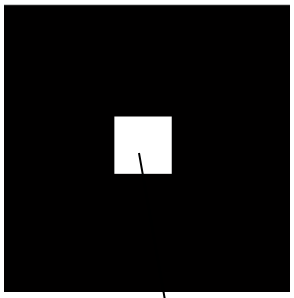
L-Shift 6

192

1	1	0	0	0	0	0	0

Extracted Second Image





If we have only 4 colors (2-bits) and we put them in the lower order bits

182

1	0	1	1	0	1	1	0

R-Shift 2

0	0	1	0	1	1	0	1

180

1	0	1	1	0	1	0	0

L-Shift 2

03

0	0	0	0	0	0	1	1

R-Shift 6

03

0	0	0	0	0	0	1	1

No need to shift right

⊕

183

1	0	1	1	0	1	1	1

Steganography

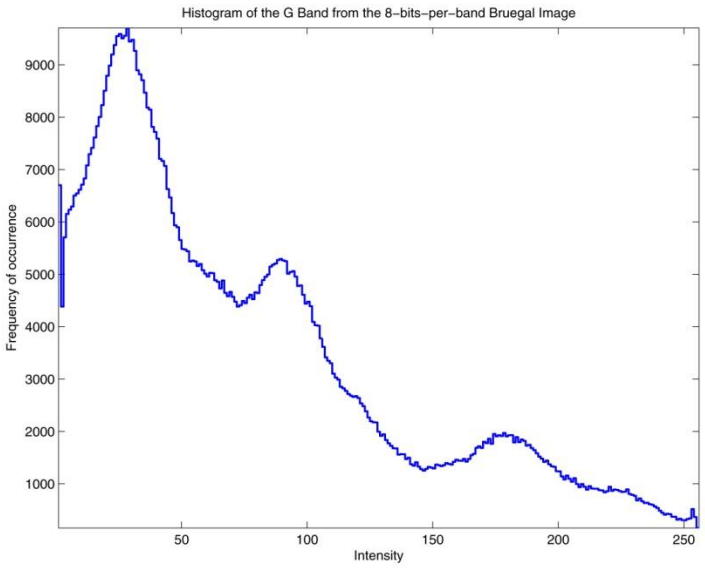
8-bit-per-band, 3-band, "original" image

Steganography

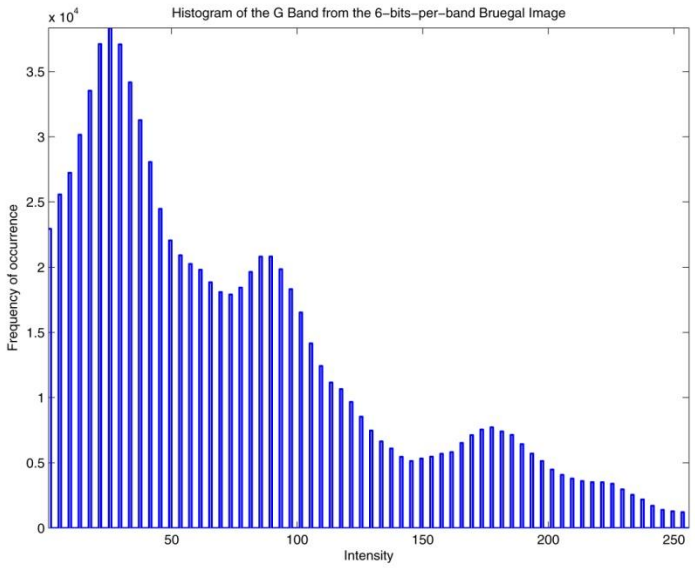
6-bit-per-band, 3-band, quantized image

Steganography

Green-band histogram of 8-bit image



Green-band histogram of 6-bit image



The histograms of the two versions indicate the difference. If the 6-bit version is displayed as an 8-bit image it has only pixels with values 0, 4, 8, … , 252.

Steganography

The second image is invisible because the value of each pixel is between 0 and 3. For any given pixel, its value is added to the to the collocated pixel in the first image that has a value from the set {0, 4, 8, … , 252}. The 2nd image is noise on the 1st.

Steganography

To recover the second image (which is 2 bits per pixel per band) simply left shift the combined image by 6 bits.

Steganography

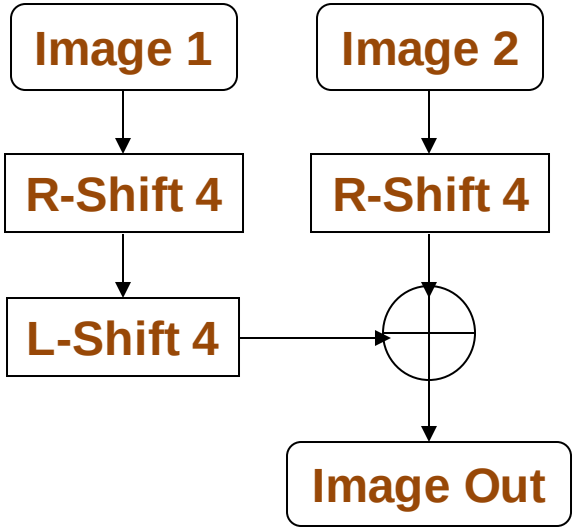
To recover the second image (which is 2 bits per pixel per band) simply left shift the combined image by 6 bits.

Steganography

This is so effective that two 4-bit-per-pixel images can be superimposed with only the image in the high-order bits visible.

Both images contain the same amount of information but the image in the low-order bits is effectively invisible

Images 1 and 2 each have 4-bits per pixel when combined.



Steganography

Original Image

Steganography

Image quantized to 4-bits per pixel.

Steganography

Image 1 in upper 4-bits.
Image 2 in lower 4-bits.

Steganography

Extracted Image

Steganography

```
im1 = rgb2gray(imread('seecs.jpg'));
im2 = rgb2gray(imread('nasa.jpg'));

im1_right_shift  = bitshift(im1,-4);
im1_left_shift  = bitshift(im1_right_shift,4);

im2_right_shift  = bitshift(im2,-4);
im2_left_shift  = bitshift(im2_right_shift,4);

figure;
subplot(2,2,1); imshow(im1);
subplot(2,2,3); imshow(im1_right_shift);
subplot(2,2,4); imshow(im1_left_shift);

figure;
subplot(2,2,1); imshow(im2);
subplot(2,2,3); imshow(im2_right_shift);
subplot(2,2,4); imshow(im2_left_shift);

combined_img = bitor(im1_left_shift,  im2_right_shift);

figure;      imshow(combined_img);
```

End

Bit-planes, Steganography